

**Modul sestav pro dopravní
monitorovací systém**

**Reporting Module for Traffic
Monitoring System**

Zadání bakalářské práce

Student: **Petr Bojko**

Studijní program: B2647 Informační a komunikační technologie

Studijní obor: 2612R025 Informatika a výpočetní technika

Téma: Modul sestav pro dopravní monitorovací systém
Reporting Module for Traffic Monitoring System

Zásady pro vypracování:

Cílem bakalářské práce je vytvoření modulu sestav pro dopravní monitorovací systém vyvíjený společností GX Solutions ve spolupráci s VŠB-TU Ostrava. Modul bude zpracovávat informace o jízdách vozidel, např. vytvářet statistiky a grafy o průměrné spotřebě, délkách jízd, nákladech na jízdu, apod. Při vytváření bakalářské práce student obdrží základní knihovny pro komunikaci s databází a dalšími existujícími částmi systému.

Jednotlivé body bakalářské práce jsou:

1. Analýza současného stavu v řešené problematice a porovnání stávajícího řešení firmy GX Solutions s dalšími existujícím softwary.
2. Vytvoření knihoven nezbytných pro generování sestav a spolupracujících s databázovými rozhraními obsahujícími potřebná data.
3. Vytvoření editovatelné tabulky knihy jízd.
4. Vytvoření grafických sestav včetně exportu.
5. Testování vytvořeného modulu s využitím reálných dat.

Seznam doporučené odborné literatury:

- [1] GX Solutions. Manuál na obsluhu SW Truck Data Memory 2010 v2.2
- [2] Martin Březina, Implementace správy reportů pro skladový systém firmy Gates Hydraulics, s.r.o., Diplomová práce, 2014

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Tomáš Kocyan**

Datum zadání: 01.09.2014

Datum odevzdání: 07.05.2015



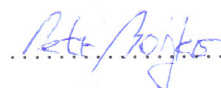
doc. Dr. Ing. Eduard Sojka
vedoucí katedry



prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 7. května 2015



Rád bych na tomto místě poděkoval především Ing. Marku Boberovi za jeho odborné rady, Ing. Petru Bučkovi za jeho všeobecné rady a podporu a Ing. Tomáši Kocyanovi za vedení mé práce a za jeho trpělivost.

Abstrakt

Práce je zaměřena na problematiku týkající se vytváření sestav a o jejich implementaci do nového monitorovacího systému společnosti GX Solutions Bohemia, s.r.o. (dále jen GX Solutions). Poté je popsána analýza konkurence GX Solutions, společně s porovnáním přítomnosti některých funkcí všech testovaných systémů. Zároveň je v práci popsáno použití reportovacích a webových nástrojů pro Microsoft Visual Studio, od společnosti Developer Express Inc. Také jsou popsány konkrétní reporty, které firma GX Solutions požadovala do svého nového systému a problémy, které se během vývoje objevily.

Klíčová slova: DevExpress, export, report, statistika, webová aplikace

Abstract

This thesis is focused on reports creation and their implementation into the new monitoring system of GX Solutions Bohemia, s.r.o. company (hereinafter GX Solutions). After that is described analyzation process of GX Solutions competition, together with presence comparison of some functions in all tested systems. In this thesis is also described work with reporting and web tools for Microsoft Visual Studio from DevExpress Inc. company. There are also described particular reports, that were demanded by GX Solutions company to their new monitoring system and problems which occurred in the development process.

Keywords: DevExpress, export, report, statistics, web application

Seznam použitých zkratk a symbolů

ACID	– Atomicity, Consistency, Isolation, Durability
AJAX	– Asynchronous JavaScript and XML
ASP	– Active Server Pages
ASPX	– Active Server Pages .NET
DDR2	– Double Data Rate 2
GB	– Gigabyte
HDD	– Hard Disk Drive
HTML	– HyperText Markup Language
IIS	– Internet Information Services
MVC	– Model View Controller
RAM	– Random-access Memory
WCF	– Windows Communication Foundation

Obsah

1	Úvod	5
2	Analýza konkurence	6
2.1	Webdispečink	6
2.2	GPSdozor	7
2.3	Fleetware	7
2.4	Carnet	7
2.5	Tabulkové porovnání	8
3	Použité technologie	10
3.1	WCF	10
3.2	ASP.NET	10
3.3	ASP.NET MVC	10
3.4	IIS Express	10
3.5	Razor syntaxe	10
3.6	DevExpress pro ASP.NET MVC	11
3.7	PostgreSQL	11
3.8	JavaScript a jQuery	11
3.9	Entity Framework	11
4	Metodika programování GX Solutions	13
4.1	Příklady pojmenování objektů	13
4.2	Struktura třídy a podobných typů	13
4.3	Obecná doporučení	13
5	Implementované sestavy	14
5.1	Obecný návrh sestav	14
5.2	Obecná implementace sestav	15
5.3	Kniha jízd	16
5.4	Přehled ujeté vzdálenosti	19
5.5	Přehled tankování	20
5.6	Přehled jízd	21
5.7	Porovnání s normou	22
6	Testování modulu	24
7	Závěr	26
8	Reference	27

Seznam tabulek

1	Tabulkové srovnání konkurence	9
2	Limity PostgreSQL	11
3	Metodika programování	13

Seznam obrázků

1	Hodnocení jízdy	6
2	Porovnání průměrné spotřeby řidičů u jednoho vozidla	6
3	Přehled překročení rychlosti	7
4	Výběr časového intervalu	8
5	Výběr časového intervalu	8
6	Výběr časového intervalu	8
7	Sdílené uživatelské rozhraní pro všechny sestavy	14
8	Stromová struktura sestav	15
9	Vizuální návrhář prvku XtraReport	16
10	Okno pro výběr exportovaných sloupců	17
11	Formulář editace tabulky Knihy jízd	17
12	Graf přehledu ujeté vzdálenosti pro jedno vozidlo	19
13	Graf přehledu ujeté vzdálenosti pro více množinu vozidel	19
14	Parametry nutné pro výpis přehledu tankování	20
15	Graf přehledu tankování	21
16	Nabídka pro výběr typu přehledu jízd	22
17	Graf zatížení CPU při výpisu záznamů za rok 2015	24
18	Graf zatížení CPU při výpisu záznamů za posledních 30 dnů	24

Seznam výpisů zdrojového kódu

1	ASPX syntaxe	10
2	Razor syntaxe pro C#	11
3	Ukázka zisku dat z databáze pro sestavu Porovnání tankování	15

1 Úvod

GX Solutions je společnost založená v roce 1998 s centrálou v Praze a provozovnou ve Frýdku-Místku. Jejím hlavním zaměřením je monitoring vozidel, především podnikových. Nejdříve poskytovala informace o využívání a stavu vozidel především u kamiónů. V současnosti mají stovky zákazníků a monitorují tisíce vozidel. Ze začátku byly informace získávány na základě off-line principu, poté, po navázání spolupráce se společností TOROLA electronic, spol. s r.o., se pustila do online monitoringu. GX Solutions poskytuje svým zákazníkům služby jako GPS monitoring, telemetrická řešení, SW a HW řešení na míru zákazníka, odborné konzultace, poradenství, školení a servisní služby.

Vize společnosti spočívá v tom, že chce v budoucnu pokračovat v inovacích řešení. Také má za cíl zobrazit důležitost telemetrie v moderních IT řešeních, bez kterých, dle jejího názoru, nelze flotilu spravovat efektivně, ať už jste v obchodě, službách, dopravě, stavebnictví nebo průmyslu.

V současné době patří mezi lídry monitoringu vozidel a strojů na českém trhu a v roce 2011 získala ocenění Nejlepší elektronický / telematický systém. Pro dispečery společností vyvinuli jak desktop, tak webovou aplikaci pro monitorování stavu vozidel. V jejich systému lze nalézt modul hlídání událostí, dispečerský modul, modul sestav a spoustu dalších funkcí.

Společnost se ale rozhodla webovou aplikaci zmodernizovat. S ohledem na ukončení podpory platformy Silverlight, na kterém běžel starý systém, se jedná o pochopitelný krok. [1]

Tato práce se zabývá modulem sestav, který je důležitou součástí nabízených služeb společnosti. Tento modul se skládá především z vytváření reportů na základě zaznamenaných dat ze snímací jednotky ve vozidle.

Pro každou dopravní společnost je velice důležité mít přehled o jízdách na svých vozidlech, o tankování a dalších statistikách spojených s jízdou, nebo náklady kolem ní. Také je důležité, aby byl modul schopen tyto věci zobrazit v přehledné formě a v případě požadavku, exportovat nebo tisknout.

Druhá kapitola bude zaměřena na analýzu konkurence. V ní budou zdůrazněny především nejzajímavější funkce, které alternativní systémy poskytují. Ve třetí kapitole budou rozepsány jednotlivé technologie, pomocí kterých systém funguje a něco málo o nich. Ve čtvrté kapitole bude zmínka o některých předních pravidlech metodiky programování GX Solutions. Pátá kapitola této práce bude popisovat návrh jednotlivých sestav a postup jejich implementace. V šesté kapitole budou popsány testy s reálnými daty, které systém podstoupil. Bude zde vypsáno i zatížení jednotky CPU na daném zařízení. V sedmé kapitole bude vypsán závěr, ve kterém bude krátce shrnuta tato práce.

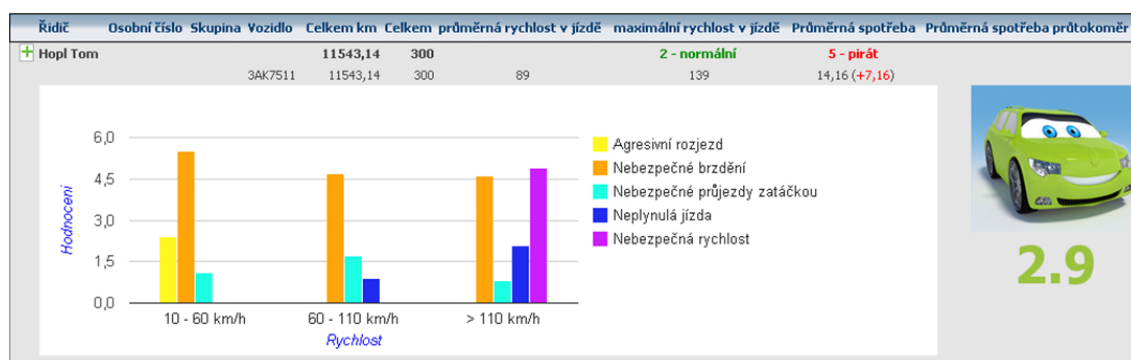
2 Analýza konkurence

Analýza konkurence byla jedním z úkolů, které bylo nutné udělat nejdříve. Analýza byla uskutečněna pomocí demo účtů, které dané společnosti nabízejí veřejnosti. Analýza probíhala v měsících červenec a srpen roku 2014. V této kapitole budou vypsány nejzajímavější funkce, které byly během analýzy objeveny, případné připomínky ke konkurenci a tabulkové porovnání přítomnosti funkcí.

2.1 Webdispečink

Pravděpodobně největší počet sestav z analyzované konkurence má Webdispečink. Najdete jich zde nespočetné množství a kvůli obrovskému počtu je možné využít i vyhledávač. Je zde také výpis naposledy použitých sestav, což je velmi dobré pro ulehčení dispečerovy práce.

Sestava, která byla u Webdispečinku zajímavá, byla sestava o hodnocení stylu jízdy řidičů. Řidič byl na základě aspektů, jako jsou třeba agresivní rozjezd nebo nebezpečné brzdění, ohodnocen známkou, která vypovídala o jeho zacházení s vozidlem. Vzhled sestavy lze vidět na obrázku 1.



Obrázek 1: Hodnocení jízdy




Další velice zajímavou funkcí, kterou by pravděpodobně dispečeri ocenili, je sestava pro porovnání spotřeby řidičů na stejném vozidle, viz obr. 2.

Řidič	Celkem km	Spotřebované PHM(l)	spotřeba průtokoměr (l)	Průměrná spotřeba (l/100km)	Průměrná spotřeba průtokoměr (l/100km)
Markéta	1421,90	146,92	0,00	10,33	0,00
Jaroslav	26,00	0,00	0,00	0,00	0,00
Karel	314,70	58,30	0,00	18,53	0,00
Celkem	1762,60	205,22	0,00	11,64	0,00

Obrázek 2: Porovnání průměrné spotřeby řidičů u jednoho vozidla

2.2 GPSdozor

Výběr sestav u GPSdozor nebyl tak obsáhlý. Funkce, která se v tomto systému narodila od systému GX Solutions objevila, byla statistika o překročení rychlosti. Dispečer zadal hodnotu v kilometrech za hodinu a byla mu vrácena sestava s výpisem událostí, kdy byla daná rychlost překročena, viz obr. 3.

<div> <div>Vytisknout</div> <div>Export do pdf</div> <div>Export do excelu</div> </div>					
Akce	Jednotka	Řidič	Max. rychl.	Rychlostní omezení	Kolikrát
	Rozvoz zboží	Jan	158 km/h	150 km/h	17
	Rozvoz zboží	Neznámý řidič	156 km/h	150 km/h	8
	Rozvoz zboží	Jaroslav	160 km/h	150 km/h	11
	Škoda Octavia	Jan	167 km/h	150 km/h	30
	Škoda Superb	Neznámý řidič	186 km/h	150 km/h	82
	VW Transporter	Neznámý řidič	186 km/h	150 km/h	57

Obrázek 3: Přehled překročení rychlosti

Systém také obsahoval editační sestavu záznamů o tankování. Sestava se skládala z velkého množství informací o provedeném tankování, včetně záznamu o natankovaném množství, způsobu platby a celkové ceně. V sumáři pod sestavou se nacházely informace o nejnižším tankování z daného období, společně s informací, kde bylo dané tankování uskutečněno.

2.3 Fleetware

V systému Fleetware se nachází velice jednoduchá, ale užitečná funkce výběru časového rozsahu, kdy stačí vybrat daný kvartál a měsíce daného kvartálu se označí, viz obr. 4.

Další velice užitečnou funkcí může být Přehled služebních jízd mimo pracovní dobu, která může být využita pro zobrazení jízd, které by měly mít označení jako soukromé, viz obr. 5.

2.4 Carnet

Největší problém u systému Carnet, co se týče sestav byl, že při každém přehození na jinou sestavu se resetoval panel pro výběr časového pásma.

c) období

Obrázek 4: Výběr časového intervalu

Vozidlo: Škoda Fabia RS (1A)
 Období: leden až březen 2014

sestava vytvořena
 2.5.2015 22:46:45

Datum	Vozidlo	Reg. značka	Řidič	Osobní číslo	Délka celkem	Aktuální NS
23.1.2014	1A	Škoda Fabia RS			96,0	
24.1.2014	1A	Škoda Fabia RS			166,0	
25.1.2014	1A	Škoda Fabia RS			277,0	

Obrázek 5: Výběr časového intervalu

Také výběr sestavy nebyl zrovna přehledný. Všechny sestavy byly zobrazeny v rozvírací nabídce, bez nějakého rozdělení na kategorie nebo abecedního seřazení. Byly pravděpodobně seřazený na základě uvážení vývojaře, které sestavy jsou nejpoužívanější.

Funkce, která se u konkurence nevyskytovala sloužila pro zobrazení zastavení vozidla v uživatelsky nadefinovaných oblastech na mapě, viz obr. 6.

Období	13	Duben	2015	Uživatelský bod	Zobrazit vše	Přepočti report	
Skupina	Zobrazit vše	Rádus [m]		Doba zastávky [min]			
RZ	Skupina vozidla	Řidič	Datum	Uživatelský bod	Evid. č. uživ. bodu	Rádus [m]	Vzdálenost od středu bodu [m]
Fabia C	DEMO/S1	demo řidič	05:27 29.04.2015	CCS		250,00	80,40
Fabia C	DEMO/S1	demo řidič	14:17 28.04.2015	CCS		250,00	80,40
Fabia C	DEMO/S1	demo řidič	16:39 24.04.2015	CCS		250,00	92,70
Fabia C	DEMO/S1	demo řidič	11:45 22.04.2015	CCS		250,00	70,80

Obrázek 6: Výběr časového intervalu

2.5 Tabulkové porovnání

V tabulce jsou shrnuty funkce, na které byla analýza nejvíce zaměřená nebo byly pro sestavy podstatné. Je zde vypsána jejich přítomnost, nebo nepřítomnost v analyzovaných systémech, viz tabulka 1.

	GX Solutions	Webdispečink	GPS Dozor	Fleetware	Carnet
Hodnocení jízdy	NE	ANO	NE	NE	NE
Srovnání spotřeby	NE	ANO	NE	NE	ANO
Překročení rychlosti	NE	ANO	ANO	NE	ANO
Měsíční přehled jízd	ANO	ANO	ANO	ANO	ANO
Export do .pdf	ANO	NE	ANO	ANO	NE
Export do .csv	ANO	ANO	ANO	NE	ANO
Export do .xls	ANO	ANO	ANO	ANO	ANO

Tabulka 1: Tabulkové srovnání konkurence

3 Použité technologie

3.1 WCF

Windows Communication Foundation je jednotný programovací model od společnosti *Microsoft*. Je vyvinut pro budování aplikací orientovaných na služby. Umožňuje vývojářům stavět bezpečné, spolehlivé a multiplatformní řešení. [3]

3.2 ASP.NET

Jedná se o webový framework, který je součástí technologie *.NET*. Tato sada knihoven umožňuje vývojáři vytvářet webové aplikace v jazyce *C#* a *Visual Basic .NET*. Knihovny obsahují již hotová řešení pro zabezpečení a šifrování, autentifikaci uživatele, práci s databází, správu formulářů a mnoho dalších. *ASP.NET* nabízí dva způsoby jak vytvářet webové aplikace. Je možné využít *ASP.NET WebForms*, kdy se webová aplikace tváří jako desktop aplikace, nebo *ASP.NET MVC*, pomocí které je vyvinuta i aplikace, o které pojednává tato práce. [4]

3.3 ASP.NET MVC

Tato technologie pracuje s konceptem, který rozděluje webovou aplikaci do tří komponent, které jsou *Model*, *Controller* a *View*. *Controller* je řídicí komponenta, která se stará o data přijatá od uživatele a předá je komponentě *Model*, která obsahuje business logiku a komunikuje s databází. Data, která *Controller* získal od *Model* pak přepoše komponentě *View*, která je zobrazí na výsledné *HTML* stránce. [4]

3.4 IIS Express

IIS Express je služba, která umožňuje vytvořit webový server, pomocí kterého lze provozovat a testovat webové stránky. Podporuje protokoly jako jsou *HTTP*, *HTTPS*, *FTP*, *FTPS*, *SMTP* a *NNTP*. [6]

3.5 Razor syntaxe

Razor syntaxe usnadňuje tvorbu *HTML* šablon za pomoci jazyka *C#* nebo *Visual Basic .NET*. Podobně jako u *ASPX* je možné zapisovat kusy kódu přímo do *HTML* stránky. [11] Funkční kód se uvozuje znakem `<%` a porovnání lze vidět na následujících úryvcích kódu. [12]

```
<ul id="products">
  <% foreach (var p in products) { %>
    <li><%=p.Name%> ($<%=p.Price%>)</li>
  <% } %>
</ul>
```

Výpis 1: ASPX syntaxe

```
<ul id="products">
  @foreach (var p in products) {
    <li>@p.Name ($@p.Price)</li>
  }
</ul>
```

Výpis 2: Razor syntaxe pro C#

3.6 DevExpress pro ASP.NET MVC

Jedná se o sadu tříd a nástrojů od společnosti Developer Express Inc, která slouží k vývoji webových aplikací. Tato sada poskytuje rozšířenou funkcionalitu ASP.NET a nástroje pro vizuální práci s komponentami. [13]

3.7 PostgreSQL

Multiplatformní systém pro správu objektově-relačních databází s otevřeným zdrojovým kódem. Splňuje všechny podmínky ACID, podporuje uložené procedury, trigger a spoustu dalších funkcí. Zahrnuje také většinu datových typů dle standardu SQL:2008. Podporuje také ukládání obrázků, zvuků i videa. [10]

Limit	Hodnota limitu
Maximální velikost databáze	Neomezená
Maximální velikost tabulky	32 TB
Maximální velikost řádku	1,6 TB
Maximální velikost pole	1 GB
Maximální počet řádků tabulky	Neomezená
Maximální počet sloupců tabulky	250 - 1600*
Maximální počet indexů tabulky	Neomezená

* závisí na typech sloupců

Tabulka 2: Limity PostgreSQL

3.8 JavaScript a jQuery

JavaScript je interpretovaný, objektově-orientovaný jazyk využívaný především pro vytváření skriptů webových stránek. Pro jednodušší vytváření webových stránek byla vyvinuta JavaScript knihovna jQuery, která ulehčuje vytváření animací, ošetřování událostí, AJAX a spoustu dalších. [8] [9]

3.9 Entity Framework

Jde o sadu technologií, které podporují vývoj aplikací orientovaných na data. Jedná se o objektově-relační mapper. Entity Framework umožňuje vývojářům pracovat s daty ve

formě doménově specifických objektů a vlastností. Díky němu se tedy nemusí potýkat přímo s tabulkami, sloupci a jejich daty. Pomocí Entity Framework jsou vývojáři schopni pracovat na vyšší úrovni abstrakce, než když se zabývají daty a mohou vytvářet aplikace pomocí méně řádků kódu, než u aplikací vyvinutých pomocí tradičních metod. [5]

4 Metodika programování GX Solutions

Společnost GX Solutions požadovala, aby při návrhu jejich systému byly dodržována určitá pravidla. Výpis těch nejdůležitějších metodik a obecných doporučení lze nalézt níže. [14]

4.1 Příklady pojmenování objektů

Typ vytvářeného objektu	Příklad pojmenování objektu	Poznámka
namespace, class, interface, event	<code>namespace Core.Database</code>	Název uvozen velkým písmenem.
delegate	<code>delegate ChangeValueDelegate</code>	Název delegáta končí slovem Delegate.
public (metody)	<code>public void GetCode()</code>	Názvy public metod začínají velkými písmeny.
const	<code>const int MAX_ROWS = 10;</code>	Konstanty vždy velkými písmeny, při více slovech oddělit pomocí podtržítka.

Tabulka 3: Metodika programování

4.2 Struktura třídy a podobných typů

1. private vlastnosti, konstanty, výčtové typy apod.
2. public vlastnosti, konstanty, výčtové typy, delegáti, události apod.
3. konstruktory
4. private metody, události, public metody apod.

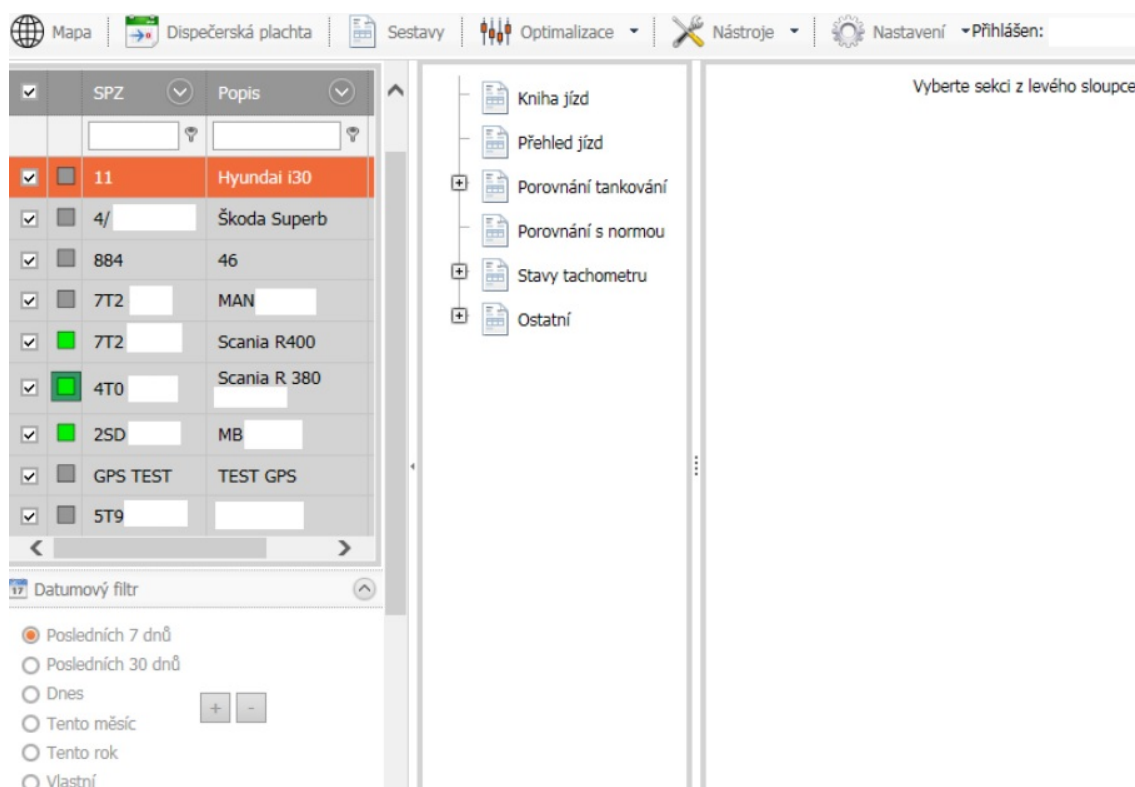
4.3 Obecná doporučení

- Je nezbytně nutné oddělovat výkonné jádro od zobrazovacího.
- Je nutné při vytváření formuláře zapínat podporu lokalizace.
- Veškeré texty a popisky v kódu je nutné ukládat do Resources a přeložit do čtyř jazyků.
- Veškeré kódy a dokumentace se musí ukládat na SVN server.

5 Implementované sestavy

V této části bude rozebrán návrh a implementace sestav.

Všechny sestavy mají společné uživatelské rozhraní. Dané rozhraní obsahuje panel, který je společný napříč všemi moduly. Tento panel umožňuje uživateli vybrat množinu vozidel, časový interval, filtry atd., viz obr. 7.



Obrázek 7: Sdílené uživatelské rozhraní pro všechny sestavy

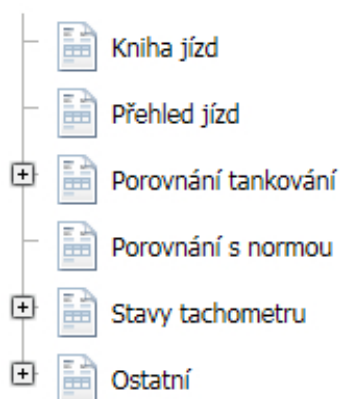
V horní části stránky se nachází nabídka, pomocí které lze přecházet mezi moduly, nástroji, nastavením apod.

Dále je zde stromová struktura, která slouží pro výběr sestavy a rozděljuje je do kategorií, viz obr. 8.

5.1 Obecný návrh sestav

Každá sestava obsahuje ve svém panelu tlačítko pro možnosti tisku sestavy. Toto tlačítko zobrazí možnosti tisku reportu ve vyskakovacím okně.

Jelikož bylo nutné danou tiskovou sestavu uživateli nejdříve zobrazit, bylo rozhodnuto o použití prvku DocumentViewer. Tento prvek, kromě samotného zobrazení, obsahuje funkce pro exportování a tisk. Tyto funkce mohou být použity prostřednictvím horního panelu prvku.



Obrázek 8: Stromová struktura sestav

Mezi nabízenými formáty jsou například .pdf, .csv, .xls, .rtf a mnohé další. U formátu .csv, v případě sestav bez grafů, proběhla implementace pro dodatečnou volbu toho, které sloupce mají být zahrnuty v exportovaném souboru.

5.2 Obecná implementace sestav

Pro vykreslení dokumentu se používá DevExpress prvek XtraReport. Základní princip je takový, že pokud uživatel klikne na tlačítko pro možnosti tisku, zavolá se JavaScript funkce, která provede operaci callback prvku CallbackPanel a zobrazí prvek PopupControl. CallbackPanel má přiřazenou funkci, která se při jeho zavolání postará o zajištění potřebných parametrů a ty pak předá Controlleru, který spustí danou funkci, v závislosti na sestavě. Tato funkce pak získá data z databáze a předá je třídě pro vykreslení reportu sestavy.

```

using (MainServiceClient client = new MainServiceClient())
{
    model.Rows = client.GetReportDataComparingFueling(model.DevicesIds,
        DataHelper.CurrentUser.CustomerId, DataHelper.CurrentUser.UserId,
        model.Since, model.Until, model.Divergence, model.IsInPercent).
        ToList();
}

```

Výpis 3: Ukázka zisku dat z databáze pro sestavu Porovnání tankování

Tato třída dědí většinu základních údajů o vzhledu reportu, jako jsou hlavička a patička dokumentu, od třídy `BaseReport`, vytvořená za pomoci vizuálního návrháře, viz obr. 9, nebo její upravená varianta v závislosti na požadavcích dané sestavy. Například, pro vykreslení dokumentu na šířku, je to třída `BaseReportLandscape`. O vykreslení detailních informací, které byly získány z databáze, se pak třída postará sama.

V kódové části vykreslovací třídy se dokument naplňuje daty. V závislosti na tom, zda se jedná o dokument s tabulkou, nebo s grafem se pro data používá buď třída `XRTable` nebo `XRChart`. Obojí součástí knihoven DevExpress.

Obrázek 9: Vizuální návrhář prvku XtraReport

Celý prvek XtraReport se pak vrací zpátky do *Controller*, kde se vloží do dočasného uložště ze kterého je předán do prvku DocumentViewer.

Export dokumentu probíhá za pomoci DevExpress funkce `ExportTo`, která jako parametr dostane prvek XtraReport.

Z důvodu nepřítomnosti funkce pro výběr exportovaných sloupců u sestav, které neslouží pro zobrazení grafů, bylo nutné ji do horního panelu prvku DocumentViewer přidat. Uživatel se po kliku na tlačítko zobrazí sloupce dané sestavy ve vyskakovacím okně. Jelikož jsou pro zobrazení vyskakovacího okna s nabízenými sloupci použity sdílené prvky `PopupControl`, `CallbackPanel` a `FormLayout`, bylo potřeba, aby tyto prvky rozpoznaly, který DocumentViewer je volal.

Toto je řešeno pomocí Javascript funkce tlačítka každého DocumentVieweru negrafové sestavy, která předá funkci od prvku `CallbackPanel` informaci o tom, odkud byla zavolána. Tato funkce se už postará o zjištění zobrazených sloupců dané sestavy. Poté informaci o sloupcích předá prvku `FormLayout`, který na jejím základě zobrazí pro každý sloupec prvek `CheckBox`, jehož odškrtnutí zapříčiní nepřítomnost sloupce ve vyexportovaném souboru, viz obr. 10.

5.3 Kniha jízd

Kniha jízd je klíčovou sestavou, díky které si může dispečer nechat vypsát jednotlivé jízdy na základě parametrů, jako jsou vybrané vozidlo a časový interval.

5.3.1 Návrh sestavy

Panel Knihy jízd zobrazí uživateli tabulku jízd s možností editace, dále pak tlačítko pro vypsání jízd do tabulky a pro možnost tisku sestavy.

☒ Datum
☒ Od
☒ Do
☒ Začátek jízdy
☒ Konec jízdy
☒ km
☒ Tachometr

Exportovat

Obrázek 10: Okno pro výběr exportovaných sloupců

Každý řádek v tabulce představuje jednu jízdu a každý sloupec její parametry, jako jsou třeba informace o času začátku a konce jízdy, počáteční a konečnou adresu, délku jízdy v kilometrech, konečný stav tachometru a další. Při kliku pravým tlačítkem na záznam tabulky se objeví kontextové menu s možností zobrazení dané jízdy v mapě.

To, které sloupce budou v tabulce zobrazeny je možno vybrat v sekci Nastavení aplikace.

Editace záznamu je umožněna při kliku na tlačítko u každého záznamu tabulky, viz obr. 11.

	2015-04-27	16:39:52	16:56:27	Radniční náměstí, Šenov, Ostrava-Město, Moravskoslezský kraj, CZE, (CZE)	Petruškova, Ostrava, Ostrava-Město, Moravskoslezský kraj, CZE, (CZE)	15,2	243 602,6
--	------------	----------	----------	--	--	------	-----------

Datum	2015-04-27	Od	16:39:52
Do	16:56:27	Začátek jízdy	Radniční náměstí, Šenov, Ostrava-Město, Moravskosl
Konec jízdy	Petruškova, Ostrava, Ostrava-Město, Moravskoslezsk	km	15,2
Tachometr	243 602,6		

✔ ✖

Obrázek 11: Formulář editace tabulky Knihy jízd

Každý sloupec obsahuje filtr, pomocí kterého lze zobrazit pouze záznamy, které daná omezení splňují.

Na posledním řádku tabulky se nachází sumář hodnot sloupců, který obsahuje hodnoty zvolené v Nastavení aplikace, pouze v případě, že je v tabulce zobrazen i jejich sloupec.

Bylo také nutné zajistit, aby při změně zařízení, nebo časového intervalu, byly záznamy v tabulce aktualizovány.

5.3.2 Implementace sestavy

Z důvodu nutnosti přítomnosti editační tabulky byl pro implementaci využit DevExpress prvek `GridView`, který tyto funkce umožňuje. Nastavení sloupců v `GridView` proběhlo za pomoci funkce `GetRowsColumnsDefinitions`, která jako parametr dostala informaci, že byla zavolána právě z Knihy jízd. Obdržené informace, které funkce vrátila, byly použity k nastavení šířky sloupců, názvů sloupců a také možnosti nebo nemožnosti editace daného sloupce.

Při spuštění panelu Knihy jízd se ze všeho nejdřív inicializuje prvek `GridView`. Poté *View*, které obsahuje dva prvky `Button`, jeden pro možnosti tisku a druhý pro výpis jízd. Dále pak jeden prvek `PopupControl`, který je zatím nastaven jako skrytý.

`GridView` má pro svou inicializaci v *Controller* metodu `DrivesContentPartialView` návratového typu `ActionResult`. Ta pomocí metody `GetLastDataDrivesForLocation` získá potřebná data z databáze. Metoda přijímá šest parametrů, které jsou následující, typ sestavy, vybrané zařízení, od, do, ID zákazníka a ID uživatele. Díky těmto parametrům nám budou vrácena potřebná data, která předáme `GridView`.

Aktualizace záznamů v `GridView` probíhá v několika případech. Pokud uživatel změní časový interval, vybrané vozidlo, nebo stiskne tlačítko pro výpis jízd. Aktualizace je zajištěna pomocí JavaScript funkce, která provede nad celým prvkem operaci `callback`, která nahraje nová data, teď již za pomoci nových parametrů.

Kontextové menu editační tabulky má pro položku Ukázat na mapě přiřazenou JavaScript metodu `onDrivesViewContextMenuClick`. Uvnitř je nejdříve otestováno, která položka z kontextového menu se má provést. V současné době je v menu jenom jedna, ale kvůli možného budoucího přidávání položek, je lepší tento test implementovat již teď. Poté metoda použije funkci `GetRowValues` na prvek `GridView` získá data řádku, z kterého bylo voláno menu. Dále je obstaráno zaslání těch získaných informací o dané jízdě metodě, která ji dále zpracuje a vykreslí do mapy.

Editace záznamu v prvku `GridView` probíhá za pomoci funkce `Update`, která se nachází v *Controller*. Hodnoty, které nám editační formulář předá, viz obr. 11, jsou typu `FormCollection`. Jelikož se během vývoje objevil problém s tím, že pokud položky dané jízdy jsou skryty, když nejsou editovatelné, tak nejsou přítomny ve `FormCollection`. Z důvodu potřeby mít ve `FormCollection` i parametry, které nejsou editovatelné, bylo nutné ho v editačním formuláři zobrazit. Místo skrytí needitovatelných prvků bylo tedy přistoupeno k zakázání jeho upravování. Prvky, které byly v *Controller* získány pomocí `FormCollection` předáme metodě `UpdateDrive`, která zároveň potřebuje informaci o vybraném zařízení, aktivním zákazníkovi a uživateli.

Pro získání dat z databáze používá funkci `GetLastDataDrivesForLocation` a ty pak předá třídě `DrivesOverviewReport`, kde jsou, narozdíl od většiny sestav, kromě vykreslení dat, dynamicky vykreslené i sloupce za pomoci funkce `GetAssignedColumns`. Vše probíhá tak, jak je výše popsáno v obecném popisu implementace sestav.

5.4 Přehled ujeté vzdálenosti

Přehled ujeté vzdálenosti je souhrn sestav, které můžou uživateli zobrazovat informace o ujetých kilometrech jak tabulkově, tak v grafu.

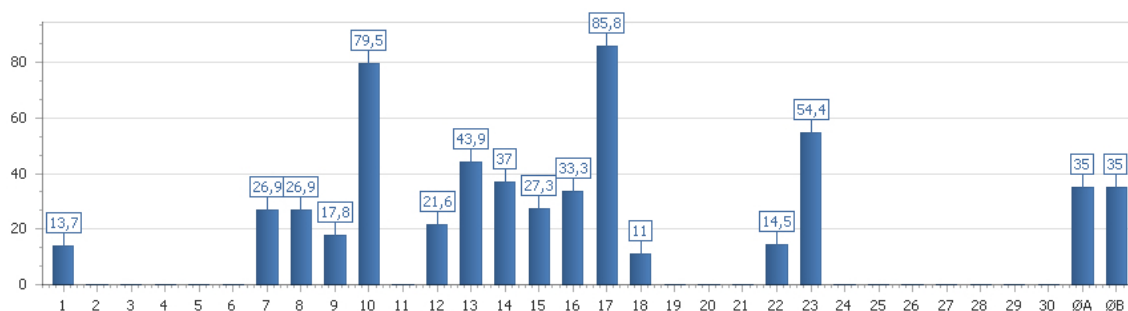
5.4.1 Návrh sestavy

Sestava slouží pro přehled ujetých vzdáleností vozidel. V případě tohoto panelu není přítomna editační tabulka, ale jenom tlačítko pro možnost tisku sestavy.

Uživatel si vybere množinu vozidel a požadovaný časový interval. Poté se klikem na tlačítko pro možnosti tisku sestavy zobrazí vyskakovací okno.

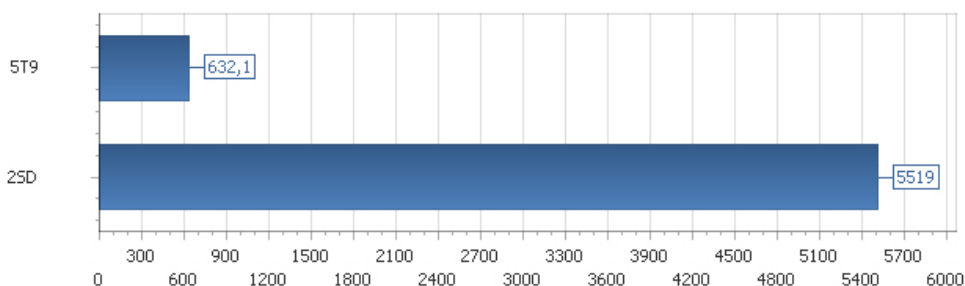
V případě tabulkové sestavy každý řádek představuje záznam o ujeté vzdálenosti jednoho vozidla a každý sloupec informace o popisu vozidla, kilometrech ujetých kvůli firemních nebo soukromých záležitostech apod. V náhledu tisku je také přítomen sumární řádek pro numerické hodnoty sestavy.

Pokud uživatel vybral sestavu Přehled ujeté vzdálenosti s grafem pro jedno vozidlo, sestava vygeneruje graf na základě ujetých vzdáleností za poslední měsíc. V tomto případě je pro každý den vypsána hodnota v kilometrech, viz obr. 12.



Obrázek 12: Graf přehledu ujeté vzdálenosti pro jedno vozidlo

Systém ale umožňuje vypsát i sestavu pro porovnání ujetých vzdáleností množiny vozidel. Graf poté zobrazuje celkovou sumu vzdáleností za daný měsíc, viz obr. 13.



Obrázek 13: Graf přehledu ujeté vzdálenosti pro více množinu vozidel

5.4.2 Implementace sestavy

Základní princip zisku dat pro zobrazení v DocumentViewer je v podstatě stejný, jako u obecné implementace sestav. Při stisku tlačítka pro možnosti tisku se zavolá callback na CallbackPanel, který data získaná pomocí metody `GetReportDataDrivingDistance`, předá do zmíněného DocumentViewer a zároveň se zobrazí, do té doby skrytý, prvek PopUpControl. Parametry, které je nutno zaslat metodě `GetReportDataDrivingDistance` jsou list zaškrnutých zařízení, ID aktivního uživatele, od a do.

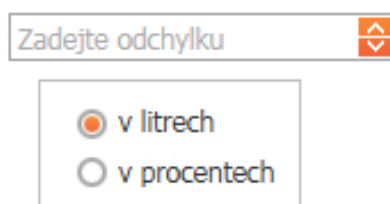
Získaná data jsou poté předána třídě `DrivingDistanceOverviewReport`. Ve třídě se data zpracují a vykreslí se pomocí třídy `XRTable`. Také dochází k počítání získaných numerických hodnot, pro jejich případné vypsatí v sumáři. U této sestavy nejsou názvy sloupců získávány z databáze, ale jsou staticky naprogramované, jelikož jsou neměnné.

V případě grafu pro ujeté vzdálenosti za daný měsíc pro jedno vozidlo se data získají pomocí funkce `GetReportDataDrivingDistanceMonthly`, s tím rozdílem, že pro časový interval se uvádí pouze datum od, ze kterého se získá informace o vybraném měsíci. Pomocí třídy `DrivingDistanceMonthlyLandscapeOverviewReport` se do grafu vykreslení údaj o ujeté vzdálenosti pro každý den ve vybraném měsíci.

U grafu pro ujeté vzdálenosti za daný měsíc pro více vozidel se pro vytažení dat použije také metoda `GetReportDataDrivingDistanceMonthly` a ve třídě pro vykreslení `DrivingDistanceMonthlyOverviewReport` se vzdálenosti jízd každého vozidla sesumarizují. Poté jsou sumarizované hodnoty vykresleny do `XRChart`, kde jedna hodnota je ujetá vzdálenost jednoho vozidla za daný měsíc.

5.5 Přehled tankování

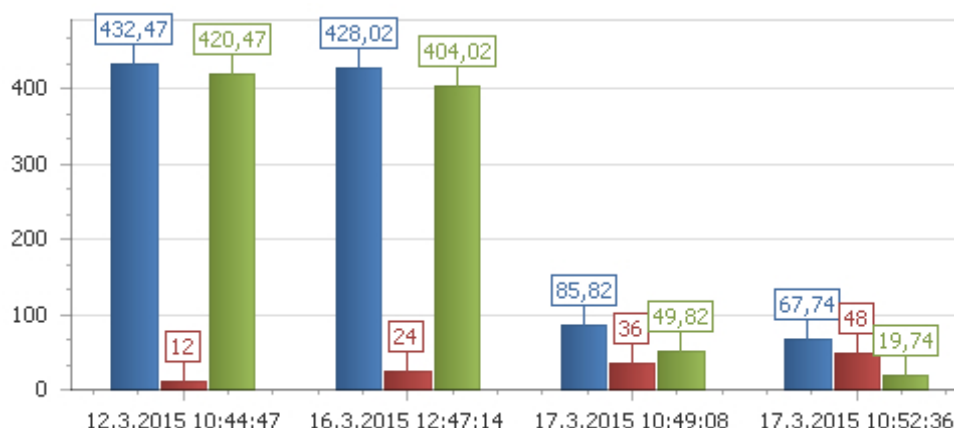
Přehled tankování umožňuje vypsat jak tabulkovou, tak grafovou sestavu s přehledem v daném časovém rozmezí pro jedno nebo více vozidel. Sestava obsahuje informace o dokladovaném a naměřeném tankování a o rozdílu mezi nimi. Tabulka obsahuje také grafický ukazatel stavu nádrže.



Obrázek 14: Parametry nutné pro výpis přehledu tankování

5.5.1 Návrh sestavy

U sestavy je nutné od uživatele získat informaci o tom, jakou odchylku mezi dokladovaným a naměřeným tankováním chce zvýraznit a v jaké jednotce bude hodnotu uvádět.



Obrázek 15: Graf přehledu tankování

Systém nabízí dvě možnosti, a to procenta a litry. Proto bylo zapotřebí implementovat prvek, který na základě vybrané jednotky bude omezovat svůj rozsah.

5.5.2 Implementace sestavy

Výše zmíněným požadavkům vyhovovaly prvky SpinEdit pro číselnou hodnotu a RadioButtonList pro volbu jednotky. Při změně jednotky byla zavolána JavaScript funkce `isInLiterOrPercentChanged`, která nastavila rozsah prvku SpinEdit. V případě, že byla vybrána procenta, byl rozsah nastaven na 0-100, v případě litrů na -9999-9999 z důvodu objemných nádrží lokomotiv, které systém může také monitorovat.

Informace získané z těchto prvků jsou pak použity jako jedny z parametrů funkce `GetReportDataComparingFueling`, která na jejich základě získá potřebná data. Tato data jsou, stejně jako u ostatních sestav, předány vykreslovací třídě, v případě, že se jedná o tabulkovou sestavu, je její název `ComparingFuelingOverviewReport`. Třída, stejně jako třída pro vykreslení tabulky Přehledu ujetých vzdáleností, má sloupce tabulky pevně naprogramované.

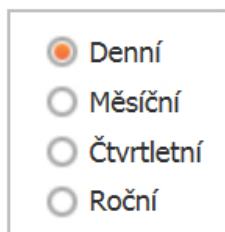
Pro vykreslení grafové sestavy je metoda pro získání dat stejná, jako v případně sestavy tabulkové, čili `GetReportDataComparingFueling`. Funkce pro samotné vykreslení je `ComparingFuelingChartOverviewReport`, která se kromě vykreslení stará i o výpočet rozdílu mezi dokladovaným a naměřeným tankováním, jelikož tato data nejsou v databázi uchovávána.

5.6 Přehled jízd

Sestava zobrazuje přehled jízd za dané časové období. Je možné si vybrat denní, měsíční, čtvrtletní a roční časové rozmezí přehledu.

5.6.1 Návrh sestavy

Sestava má jednoduchý panel, který obsahuje pouze tlačítko pro možnosti tisku a prvek, ze kterého si uživatelé mohou vybrat časové rozmezí pro přehled jízd, viz obr. 16.



Obrázek 16: Nabídka pro výběr typu přehledu jízd

5.6.2 Implementace sestavy

Jelikož bylo nutné vybrat prvek, který umožňuje volbu pouze jedné možnosti, znovu padla volba na prvek `RadioButtonList`. Informace o vybrané hodnotě pak byla předána metodě `GetReportDataSumDrives`, společně s informací o vybraném zařízení a dalšími parametry. Navracená data byla poté předána vykreslovací třídě `SumDrivesOverviewReport`.

Ta má na starost vykreslení dat do tabulky, stejně jako *Knihy jízd*, má ale navíc zvlášť tabulku pro obecné informace o datech z tabulky. V ní mohou být data sumarizovaná, zprůměrovaná, nebo může být uvedena jejich poslední hodnota. Všechny tyto informace byly získány z databáze na základě nastavení, získaného ze sekce Editoru sloupců v Nastavení aplikace, aktivního uživatele.

Sloupce tabulky jsou vykreslovány automaticky, na základě údajů z databáze, stejně jako v případě *Knihy jízd*.

5.7 Porovnání s normou

Porovnání s normou umožňuje uživateli porovnat skutečné naměřené údaje za pomoci monitorovací jednotky, jako jsou například průměrná spotřeba nebo spotřeba při použití topení, s údaji normovanými, které většinou uvádí výrobce zařízení.

5.7.1 Návrh sestavy

Uživatel nebude zadávat žádné dotatečné vstupy, kromě vybraného zařízení a časového rozpětí v levém panelu. Proto pro panel této sestavy bude zobrazeno jenom tlačítko pro zobrazení možností tisku.

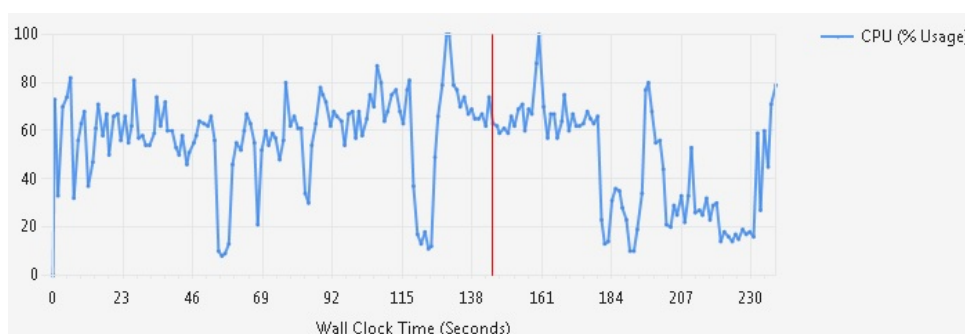
5.7.2 Implementace sestavy

Základ sestavy byl naprosto stejný, jako u většiny sestav. Pro získání dat z databáze byla využita metoda `GetReportDataComparingWithNorm` a získaná data vykreslila do tabulky

metoda `NormComparisonOverviewReport`. Tato sestava má také neměnné sloupce, proto byly ve třídě naprogramovány na pevno.

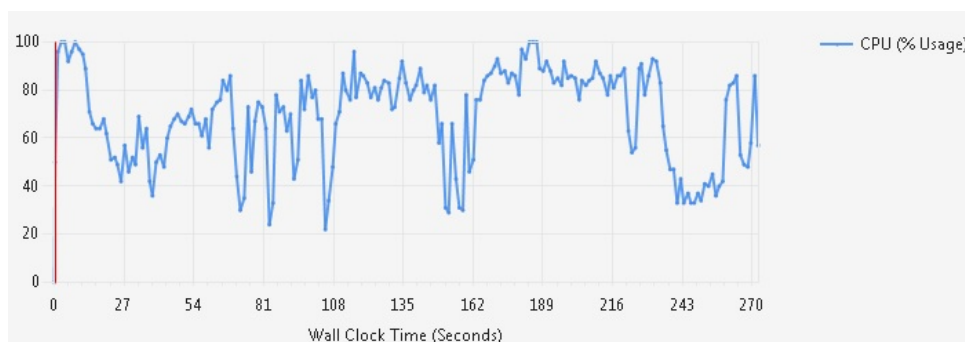
6 Testování modulu

Modul byl podroben testům za pomoci reálných dat získaných z databáze GX Solutions. Při testech bylo měřeno zatížení jednotky CPU. Nejdříve se modul testoval pomocí vyplnění prvku GridView daty, za celý rok 2015, od startu systému, až po dokončení vyplnění, viz obr. 17.



Obrázek 17: Graf zatížení CPU při výpisu záznamů za rok 2015

Poté test proběhl znovu nad stejnou operací s výjimkou toho, že se vypisovala data za posledních 30 dnů, viz obr. 18.



Obrázek 18: Graf zatížení CPU při výpisu záznamů za posledních 30 dnů

Vytížení jednotky CPU je větší u druhého testu. Toto může být zapříčiněno vlivem ostatních modulů. Samotná funkce `GetLastDataDrivesForLocation`, která se stará o vytážení dat z databáze, totiž v prvním případě vytížila CPU na 2,68%, v druhém případě pouze na 0,94%.

Test byl proveden na zařízení Asus K61IC. Níže lze nalézt hlavní parametry zařízení.

- Intel Core 2 Duo T5870
- NVidia GT220M 2GB
- 4GB RAM DDR2 800 MHz

- 500GB HDD

7 Závěr

Úkolem této práce bylo vytvoření sestav pro nový monitorovací systém GX Solutions. Tyto sestavy budou po nasazení do ostrého provozu sloužit pro potřeby dispečerů dopravních firem.

Díky jejich funkcím si budou uživatelé schopni vypsát informace spojené se záznamenými údaji o činnosti zařízení společnosti. Tyto údaje budou moci prostřednictvím modulu sestav schopni také vytisknout nebo vyexportovat do několika různých formátů.

Během testování modulu pomocí reálných dat nedošlo k žádným problémům. Je to také díky tomu, že většina sestav nepotřebuje příliš mnoho vstupů od uživatele.

8 Reference

- [1] *Fáze životního cyklu produktu* [online]. 2015 [cit. 2015-05-05]. Dostupné z: <https://support.microsoft.com/cs-cz/lifecycle?C2=12905>
- [2] *O Firmě | GX Solutions* [online]. 2015 [cit. 2015-05-05]. Dostupné z: <http://www.gxsolutions.cz/o-firme/>
- [3] *Windows Communication Foundation* [online]. 2015 [cit. 2015-05-05]. Dostupné z: <https://msdn.microsoft.com/en-us/library/dd456779%28v=vs.110%29.aspx>
- [4] *1. díl - Úvod do ASP.NET* [online]. 2015 [cit. 2015-05-05]. Dostupné z: <http://www.itnetwork.cz/tutorial-uvod-do-asp-dot-net>
- [5] *Entity Framework* [online]. 2015 [cit. 2015-05-05]. Dostupné z: [https://msdn.microsoft.com/en-us/library/gg696172\(v=vs.103\).aspx](https://msdn.microsoft.com/en-us/library/gg696172(v=vs.103).aspx)
- [6] *IIS Express Overview : The Official Microsoft IIS Site* [online]. 2015 [cit. 2015-05-05]. Dostupné z: <http://www.iis.net/learn/extensions/introduction-to-iis-express/iis-express-overview>
- [7] *Notebooks & Ultrabooks - K61IC - ASUS Global* [online]. 2015 [cit. 2015-05-05]. Dostupné z: http://www.asus.com/Notebooks_Ultrabooks/K61IC/specifications/
- [8] *JavaScript reference - JavaScript - MSDN* [online]. 2015 [cit. 2015-05-05]. Dostupné z: <https://developer.mozilla.org/cs/docs/Web/JavaScript>
- [9] *jQuery* [online]. 2015 [cit. 2015-05-05]. Dostupné z: <https://jquery.com/>
- [10] *PostgreSQL: About* [online]. 2015 [cit. 2015-05-05]. Dostupné z: <http://www.postgresql.org/about/>
- [11] *ASP.NET Razor C# Syntax* [online]. 2015 [cit. 2015-05-05]. Dostupné z: http://www.w3schools.com/aspnet/razor_syntax.asp
- [12] *asp.net - Comparisons of Razor vs ASPX syntax - Stack Overflow* [online]. 2015 [cit. 2015-05-05]. Dostupné z: <http://stackoverflow.com/questions/9699556/comparisons-of-razor-vs-aspx-syntax>
- [13] *Online Documentation - Developer Express Inc.* [online]. 2015 [cit. 2015-05-05]. Dostupné z: <https://documentation.devexpress.com/#AspNet/CustomDocument7873>
- [14] [online]. Marek Bober, poslední aktualizace 28.6.2014, verze 4.0, 2015 [cit.2015-05-06]. Dostupné z: SVN server